

## I bims, 1 String!

### Aufgabe 1. Indizes.

- Gegeben ist die Zeichenkette „I bims, 1 String!“. Schreiben Sie die Zeichenkette zeichenweise in die Zeile II der Tabelle.
- Ergänzen Sie in Zeile I der Tabelle die Indizes (Pl. von Index) der Zeichen im String.
- Ergänzen Sie die negativen Indizes der Zeichen im String in Zeile III der Tabelle.

I																	
II																	
III																	

### Aufgabe 2. len().

Gegeben ist dieses Python-Skript.

```
1 zk = "I bims, 1 String!"
2 a = zk[0]
3 b = zk[-13]
4 c = len(zk)
5 print(a+b)
```

- Geben Sie an, was das Programm ausgibt.
- Nennen Sie den Wert der Variable c. Begründen Sie Ihre Antwort.
- Nennen Sie den Datentyp der Variable c.

### Aufgabe 3. Programm. (Hinweis: <https://l.phina.be/1>)

Schreiben Sie ein Programm, das

- als Nutzer:innen-Eingabe eine Zeichenkette und zwei Zahlen hat. Die Zahlen sollen als Indizes für die Zeichen im String dienen.
- prüft, ob beide Indizes zu gleichen Buchstaben passen (wie der 2. und letzte Buchstabe im Wort „Mississippi“: beides „i“). Gibt es eine Übereinstimmung, so soll „gleicher Buchstabe“ ausgegeben werden. Gibt es keine Übereinstimmung, so soll „verschiedene Buchstaben“ ausgegeben werden.
- Zusatz:** prüft, ob beide Indizes auf denselben Buchstaben referenzieren (wie z.B. der 4. und der letzte Buchstabe des Wortes „Hamm“: sie verweisen beide auf das 2. „m“). Fällt die Überprüfung positiv aus, soll „Derselbe Buchstabe,“ ausgegeben werden, sonst „Nicht derselbe.“ Hinweis: diese Überprüfung muss mathematisch erfolgen.

### Aufgabe 4. Slices.

Python erlaubt es auch, Teile von Zeichenketten zu referenzieren. Markieren Sie in der rechten Spalte, welche Zeichen ausgegeben werden (Hinweis: Das Zeichen `_` zeigt ein Leerzeichen an.). Überprüfen Sie Ihre Vermutung.

```
1 zk = "I bims, 1 String!"
2 print(zk[2:5])
3 print(zk[:9])
4 print(zk[10:])
```

I\_bims,\_1\_String!

I\_bims,\_1\_String!

I\_bims,\_1\_String!

### Aufgabe 5. Aus klein mach groß und umgekehrt. (Hinweis: <https://l.phina.be/1>)

Folgende Aufgabe können Sie nur mit `ord()` und `chr()` lösen. Schreiben Sie ein Programm, das als Eingabe einen einzelnen Buchstaben hat. Wird eine längere Zeichenkette eingegeben, soll nur das erste Zeichen bearbeitet werden. Ist das Zeichen ein Kleinbuchstabe, soll das Programm das Zeichen als Großbuchstaben ausgeben und umgekehrt. Ist das Zeichen kein Buchstabe, soll „Kein Buchstabe“ ausgegeben werden.